Semantic Voxel Grid Mapping for Indoor Environments

Faris Hajdarpasic Ajay Ragh Prasanna Bijja Haofei Kuang

Abstract— **Mapping is an integral part of any mobile robotics application. It is required to efficiently represent the scene around the robot and is important for applications like SLAM, localization, and path planning. Traditional mapping techniques aim to represent the geometric information about the scene in the map representation. With the development of deep learning networks, it has become increasingly easier to generate semantic segmentation from RGB and depth data. In this project, we aim to incorporate this semantic information into a voxel grid map representation to enhance the information encoded in the map.**

I. Introduction

Map construction of the environment is an essential part of mobile robot systems. Whether the robot is operating in an indoor or outdoor environment, a map of that environment is needed as the input for further tasks. Based on the map, the robot can localize itself, plan the path to a certain goal, navigate through the environment, and do many other tasks to support its autonomy. In general, maps are the main building block of the autonomous robots pipeline.

One of the ways to represent the environment using 3D representation. There are different kinds of 3D map representations, such as point clouds, voxel grid maps, and surfel [13] based maps. Commonly, all of these maps can be used to represent the geometry of the environment.

In addition to having only geometrical representations of the environment, we aim to incorporate semantic information into a purely geometric representation, thus providing more information about the environment. This additional knowledge about the scene can be very useful for tasks such as localization or planning.

II. Related Work

With the development of semantic segmentation networks, it became easier to easily obtain pixel-level semantic segmentation of RGB images. We explored multiple existing techniques that provide semantic segmentation of input RGB data like ESANET [11] [12], SSMA [15], FuseNet [5], and so on. These networks are capable of taking in RGBD data and generating semantic images from them.

There are currently various existing approaches to represent the physical world around a robot. These include occupancy grid maps, point clouds, voxel grids, etc. The concept of voxel grids being used for representing the map of the scene is a common approach and can be seen in works like [7] [8]. Voxel grids allow to discretize the world into voxel grids of controllable resolution which enables efficient representation.

The idea of semantic mapping has been explored in works like [18] [17] [2] where object-level semantic information is used for constructing semantic maps for long-term indoor localization. These works have proven how semantic addition of semantic information can improve further processing in mobile robots. The work done in [1] explores a method of multi-class recursive Bayesian approach that allows to generation of probabilistic maps incorporating multiple semantic class probabilities in a way similar to binary recursive Bayesian occupancy grid mapping done in works like [4].

III. Our Approach

Our approach consists of using RGBD images, corresponding poses, and semantic information obtained from the semantic segmentation network. The technique consists of creating a 3D point cloud from RGBD images into the global frame, sub-sampling point cloud using a voxel-based approach, and using a recursive Bayesian Filter to determine the semantic class of the voxel.

A. Input

As mentioned, the input to our pipeline consists of time-synchronized RGB, depth images, and pose information. The RGB and depth images are passed through a semantic segmentation network to obtain the semantic probabilities for each pixel.

Semantic segmentation networks like ESANET [11] [12], and SSMA [15] provide pixel-wise semantic labels as the output. We require the pixel-wise semantic class probabilities instead. To achieve this we take the output of the network just before the final layer where the argmax function is applied, which returns the label with the highest probability for that pixel.

The pose data can be obtained from SLAM algorithms like RTABMAP [6], GMapping [4] etc which provide robust robot poses. Once we have the semantic probabilities and the corresponding poses of the robot and then we can proceed to project the pixels into the global frame and then use our approach to estimate the semantic labels of the corresponding location in the global frame.

B. Map Estimation

Once the pixels are projected into the global frame we have a point cloud representation of the scene. Now we subsample the point cloud into voxel representation for efficient processing. Once the voxel cloud is generated we move towards estimating the semantic classes.

Each pixel in the input data is projected into the global frame. Since voxels are a discrete representation of the scene, we can have multiple pixels that project into the same voxel. As the robot moves through the scene we will have more images which will also have pixels projecting into existing voxels.

Traditional recursive Bayesian-based mapping techniques work on probability distributions that deal with a binary case of a cell being occupied or free. The work done by [1] explores a multi-class variant of this approach where we can use a recursive Bayesian method to update probability distributions representing multiple classes. Figure 1 represents how we use this approach to update the semantic probabilities of the voxels during our map estimation. For each pixel, once we project them into the global frame using the depth, camera extrinsic and robot pose we will be able to calculate the coordinates of the voxel to which it belongs. Once we identify the voxel then we update the multi-class probability distribution.

The posterior of the map can be calculated by applying Bayes' rule.

$$p_{t+1}(\mathbf{m}) \propto p(Z_{t+1}|\mathbf{m}, \mathbf{X_{t+1}})p_t(\mathbf{m}) \tag{1}$$

After mathematical derivation, cell m_i posterior in log-odds space ($\mathbf{h}$), can be calculated as following:

$$\mathbf{h}_{t+1,i} = \mathbf{h}_{t,i} + \sum_{\mathbf{z}\in Z_{t+1}} (\mathbf{l_i}(\mathbf{z}) - \mathbf{h_{0,i}}) \tag{2}$$

where

$$\mathbf{h}_{t,i} = \left[\log\frac{p_t(m_i=0)}{p_t(m_i=0)} \cdots \log\frac{p_t(m_i=K)}{p_t(m_i=0)}\right] \in \mathbb{R}^{K+1} \tag{3}$$

and log-odds of the inverse sensor model is:

$$\mathbf{l}_i = \left[\log\frac{p_t(m_i=0)|\mathbf{z}}{p_t(m_i=0)|\mathbf{z}} \cdots \log\frac{p_t(m_i=K)|\mathbf{z}}{p_t(m_i=0)|\mathbf{z}}\right] \tag{4}$$

Since we are constructing a point cloud from RGBD images, we already know what points correspond to what voxels, and therefore we update belief of that voxel using the following inverse sensor model:

$$\mathbf{l}(\mathbf{z}) = \log\text{ odds(class probabilities)} \tag{5}$$

where the pivot class is arbitrarily chosen.

C. Output

The map estimation step updates the robot's belief about the scene and its semantic information as the robot moves through the environment. The semantic class with the highest probability is chosen as the semantic label for each voxel. We will have a 3D semantic voxel cloud as the output map representation from our pipeline. This semantic map can then be used for further processing like path planning, localization, etc. Figure 2 shows a semantic voxel grid we generated using this approach using data collected from Habitat simulator [10] [14] [9].

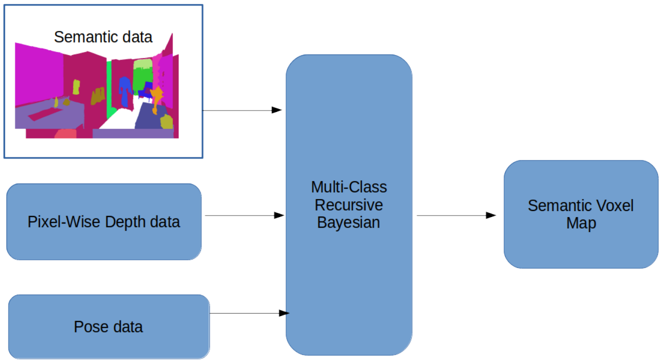

Fig. 1: Mapping pipeline

IV. Experiments

The main focus of this work is a semantic mapping technique that will generate a 3D voxel map representation of the world, which in addition to providing geometric information, also provides semantic understanding of the scene. To demonstrate the working of our pipeline we performed the following experiments. For ease of development, instead of using a semantic segmentation network, we used AI habitat simulator to simulate an indoor scene from the Matterport semantic dataset. The simulator is capable of simulating a 'semantic sensor' that provides us with semantic information about the scene along with synchronized RGBD images and pose data.

A. Experimental Setup

We collected data from the above-mentioned Matterport [16] scene by navigating a simulated agent through it in the Habitat simulator. Once the data is collected we then process this data to be used as input for our pipeline. The Matterport dataset consists of 40 semantic classes. The semantic sensor provides us with instance-level labels of the objects in each room.

B. Mapping

We first mapped these instance labels into the 40 semantic classes represented in the dataset. Once this mapping is done we have to now generate semantic probabilities for each pixel. To achieve this we first created an array representing the 40 classes for each pixel. Now in this array, we assign the class represented by the semantic label with a high probability value of 0.9. For the remaining classes, we spread evenly the remaining probability of 0.1. So once this processing is done we will have an $(N, 40)$ array where N is the number of pixels and 40 is the semantic class.

The pixels are then projected into the world frame using the depth and camera pose which will be represented as an array of shapes $(N, 3)$. Where N is the number of pixels and 3 represents the x,y,z coordinates in the global frame. These

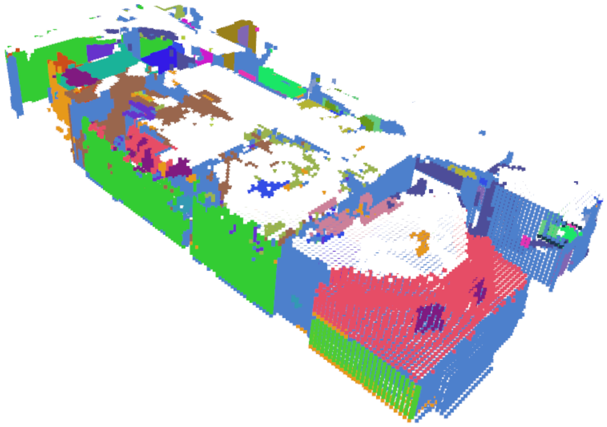

Fig. 2: Voxel grid map created from Matterport scene

coordinates are then discretized to obtain the final voxel cell coordinates in the global frame.

Once we have this information now we can perform our recursive Bayesian update to update the probability distributions of the voxel cells. Finally, once all the input data is processed we perform an $argmax$ operation over each voxel cell to decide its semantic label. This map representation can be used for additional processing.

C. Localization

We performed an additional implementation of a Monte Carlo localization [3] which is a particle filter-based approach for localization. Within this, we implemented a semantic label-based particle weight update for the observation model. Our approach requires a prior available map representation (i.e. ground truth map), of the entire scene generated using our pipeline. This map is in the same global frame as the particles.

Now during each iteration of the particle filter, for each particle, we use their pose in the global frame and the corresponding RGBD images and semantic labels of the iteration, we generate a map in the global frame using our pipeline (local map of a particle). Now we create an array of semantic labels extracted from each voxel cell coordinate of the local map. Similarly from the global map we extract the semantic labels at the same voxel cell coordinates represented in the local map. Once we have these two arrays we perform a cosine similarity over these two arrays and use the output as the particle weight.

Since this operation has to be done for each particle, as of now, we are not able to perform a full-fledged particle filter using a large number of particles due to the huge processing times required. Instead, we were only able to test this approach using a very small number of particles. As of now, we have not been able to obtain promising results. We believe this is an area that can be investigated more.

V. Conclusion

In this project, we explored an approach to incorporate semantic information into existing map representations that generally only contain geometric information about the scene around the robot. Our approach uses a multi-class recursive Bayesian approach to generate a semantic voxel grid map of the scene. We implemented our approach and tested it using data collected from a simulated indoor environment in AI habitat simulator. We were able to obtain a map that represents the environment and provides semantic information about the scene around it.

There is further space for evaluating quality of this semantic map, and improving localization that uses this map.

References

[1] A. Asgharivaskasi and N.A. Atanasov. Active bayesian multi-class mapping from range and semantic segmentation observations. *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1–7, 2021.

[2] Y. Dehbi, L. Klingbeil, and L. Plümer. Uav mission planning for automatic exploration and semantic mapping. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2020.

[3] F. Dellaert, D. Fox, W. Burgard, and S. Thrun. Monte carlo localization for mobile robots. *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No.99CH36288C)*, 2:1322–1328 vol.2, 1999.

[4] G. Grisetti, C. Stachniss, and W. Burgard. Improved techniques for grid mapping with rao-blackwellized particle filters. *IEEE Transactions on Robotics*, 23(1):34–46, 2007.

[5] C. Hazirbas, L. Ma, C. Domokos, and D. Cremers. Fusenet: incorporating depth into semantic segmentation via fusion-based cnn architecture. In *Asian Conference on Computer Vision*, November 2016.

[6] M. Labbé and F. Michaud. Rtab-map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation. *Journal of Field Robotics*, 36(2):416–446, 2019.

[7] M. Muglikar, Z. Zhang, and D. Scaramuzza. Voxel map for visual slam. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4181–4187, 2020.

[8] T.T. Nguyen, P. Nguyen, C.H. Nguyen, and M. Tran. Examination of sampling-based path planning for indoor uav using voxel grid-based visual slam. *Proceedings of the 8th International Conference on Robotics and Artificial Intelligence*, 2022.

[9] X. Puig, E. Undersander, A. Szot, M.D. Cote, R. Partsey, J. Yang, R. Desai, A.W. Clegg, M. Hlavac, T. Min, T. Gervet, V. Vondrus, V.P. Berges, J. Turner, O. Maksymets, Z. Kira, M. Kalakrishnan, J. Malik, D.S. Chaplot, U. Jain, D. Batra, A. Rai, and R. Mottaghi. Habitat 3.0: A co-habitat for humans, avatars and robots, 2023.

[10] M. Savva, A. Kadian, O. Maksymets, Y. Zhao, E. Wijmans, B. Jain, J. Straub, J. Liu, V. Koltun, J. Malik, D. Parikh, and D. Batra. Habitat: A Platform for Embodied AI Research. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.

[11] D. Seichter, M. Köhler, B. Lewandowski, T. Wengefeld, and H.M. Gross. Efficient rgb-d semantic segmentation for indoor scene analysis. *arXiv preprint arXiv:2011.06961*, 2020.

[12] D. Seichter, M. Köhler, B. Lewandowski, T. Wengefeld, and H.M. Gross. Efficient rgb-d semantic segmentation for indoor scene analysis. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 13525–13531, 2021.

[13] J. Stückler and S. Behnke. Multi-resolution surfel maps for efficient dense 3d modeling and tracking. *J. Vis. Commun. Image Represent.*, 25:137–147, 2014.

[14] A. Szot, A. Clegg, E. Undersander, E. Wijmans, Y. Zhao, J. Turner, N. Maestre, M. Mukadam, D. Chaplot, O. Maksymets, A. Gokaslan, V. Vondrus, S. Dharur, F. Meier, W. Galuba, A. Chang, Z. Kira, V. Koltun, J. Malik, M. Savva, and D. Batra. Habitat 2.0: Training home assistants to rearrange their habitat. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

[15] A. Valada, R. Mohan, and W. Burgard. Self-supervised model adaptation for multimodal semantic segmentation. *International Journal of Computer Vision (IJCV)*, jul 2019. Special Issue: Deep Learning for Robotic Vision.

[16] K. Yadav, R. Ramrakhya, S.K. Ramakrishnan, T. Gervet, J.A. Turner, A. Gokaslan, N. Maestre, A.X. Chang, D. Batra, M. Savva, A.W. Clegg, and D.S. Chaplot. Habitat-matterport 3d semantics dataset. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4927–4936, 2022.

[17] N. Zimmerman, T. Guadagnino, X. Chen, J. Behley, and C. Stachniss. Long-term localization using semantic cues in floor plan maps. *IEEE Robotics and Automation Letters*, 8:176–183, 2022.

[18] N. Zimmerman, M. Sodano, E. Marks, J. Behley, and C. Stachniss. Constructing metric-semantic maps using floor plan priors for long-term indoor localization. *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1366–1372, 2023.